Chronic Pain Estimation Through Deep Facial Descriptors Analysis *

Antoni Mauricio[1,2][0000−0003−3279−7563], Jonathan Peña[2], Erwin Dianderas[3], Leonidas Mauricio[2], Jose Díaz[2][0000−0002−8372−7760], and Antonio Morán[2]

[1] Department of Computer Science
Universidad Católica San Pablo, Arequipa, Perú
manasses.mauricio@ucsp.edu.pe
[2] Universidad Nacional de Ingeniería, Lima, Perú
jonathan.pena.a@uni.pe,amoran@ieee.org
jcdiazrosado@uni.edu.pe
[3] Instituto De Investigaciones De La Amazonía Peruana, Perú
erwin.dianderasc@ciplima.org.pe

Abstract. Worldwide, chronic pain has established as one of the foremost medical issues due to its 35% of comorbidity with depression and many other psychological problems. Traditionally, self-report (VAS scale) or physicist inspection (OPI scale) perform the pain assessment; nonetheless, both methods do not usually coincide [14]. Regarding self-assessment, several patients are not able to complete it objectively, like young children or patients with limited expression abilities. The lack of objectivity in the metrics draws the main problem of the clinical analysis of pain. In response, various efforts have tried concerning the inclusion of objective metrics, among which stand out the Prkachin and Solomon Pain Intensity (PSPI) metric defined by face appearance [5]. This work presents an in-depth learning approach to pain recognition considering deep facial representations and sequence analysis. Contrasting current state-of-the-art deep learning techniques, we correct rigid deformations caught since registration. A preprocessing stage is applied, which includes facial frontalization to untangle facial representations from non-affine transformations, perspective deformations, and outside noises passed since registration. After dealing with unbalanced data, we fine-tune a CNN from a pre-trained model to extract facial features, and then a multilayer RNN exploits temporal relation between video frames. As a result, we overcome state-of-the-art in terms of average accuracy at frames level (80.44%) and sequence level (84.54%) in the UNBC-McMaster Shoulder Pain Expression Archive Database.

Keywords: CNN-RNN Hybrid Architecture, Pain Recognition, Deep Facial Representations.

* The present work was supported by grant 234-2015-FONDECYT (Master Program) from Cienciactiva of the National Council for Science, Technology and Technological Innovation (CONCYTEC-PERU) and the Vicerrectorate for Research of Universidad Nacional de Ingeniería (VRI - UNI).

1 Introduction

According to Raffaeli et al. [9], chronic pain is one of the most complex medical ills, treated in multiple ways depending on its severity. In 2011, 20% of adults worldwide suffered at least one kind of pain, and some estimations point that around 10% of adults would be diagnosed with chronic pain each year. In 2017, Souza et al. [12] documented that patients diagnosed with chronic pain rose to 50%, although distribution varies per region due to many factors, including life quality and stress. Pain feeling is a personal experience, which can be expressed to the medics using a Visual-Analogue-Scale (VAS) or estimated by a standardized Observed Pain Intensity Scale (OPI). Nevertheless, both metrics tend not to coincide because patients could misexpress the pain or misinterpret by medics [14].

In the computational field, automatic pain recognition has received increased attention since Lucey et al. [4] published the UNBC-McMaster Shoulder Pain Expression Archive Database. The problem has been intensely explored either for computer vision classical methods [2, 3, 8, 10, 15] or brand-new deep learning approaches [6, 11, 16]. The UNBC-McMaster database consists of 200 video clip sequences taken from 25 patients who were suffering from shoulder pain. It is labeled both at frames (PSPI on a range of [0-15]) and sequences level (VAS on a scale of [0-10] and OPI on a range of [0-5]). Classic proposals have shown outstanding results through geometrical analysis but almost limited to the frame level analysis. However, even though temporal analysis allows us to explore the problem deeply, it does not present such remarkable results. This result seems to be caused by the vanishing gradient problem, which is proportional to the sequence size [13].

In this paper, we aim to solve two issues: (1) untangle the facial representations between aspects and facial pain expression; and, (2) prevent the vanishing gradient problem in temporal analysis. To accomplish them, we propose two procedures. First, a preprocessing stage that considers facial landmarks for masking and frontalization. Second, a CNN-RNN hybrid model which is designed to surpass the vanishing gradient problem by using low-processing sequential units. The upcoming sections go as follows: Section 2 covers the current state-of-the-art related to pain analysis. Section 3 exposes the detail of our proposals. Section 4 includes the most relevant experiments and their respective results contrasted with literature. Finally, we discuss our findings and conclusions in Section 5, including future improvements.

2 Related Work

Like most current computer vision applications, research papers can be grouped based on its features extraction approach. Table 1 presents a summary of multi-class classification research works, including their metrics and results. Florea et al. [2] propose to transfer the pivotal face features extracted from the Cohn-Kanade (CK+) dataset using a Histogram of Topographical Features (HoT) to

a two-levels-of-classification model based on Support Vector Regressors (SVR). Rathee et al. [10] define the facial deformations using rigid and non-rigid parameters by Thin-Plate Spline (TPS) mapping. Moreover, they mapped the deformation parameters to higher discriminative space using the Distance Metric Learning (DML) method and use an SVM to carry out the 16-classes classification.

	Classifier	**Features**	**Details**		**Pain**	**Metric**	**Score**
			Ba	**Pp**	**Levels**		
Zhao et al. [15]	OSVR-L2	LBP + Gabor	x	x	[0-5]	PCC MAE ICC	0.60 0.81 0.56
Kaltwang et al. [3]	RVM	DCT + LBP	-	-	[0-15]	MSE PCC ICC	1.39 0.59 0.50
Florea et al. [2]	SVR	HoT	-	-	[0-15]	MSE PCC	1.18 0.55
Rathee et al. [10]	SVM	TPS + DML	x	-	16	ACC	0.96
Zhou et al. [16]	RCNN	-	x	x	16	MSE PCC	1.54 0.65
Nasrollahi et al. [6]	CNN-LSTM	-	-	x	14	ACC	0.619
Rodriguez et al. [11]	CNN-LSTM	-	x	x	5	MAE MSE PCC ICC	0.5 0.74 0.78 0.45

Table 1. Metrics comparison of the-state-of-the-art. The table includes the classifier description, features selection, implementation details, metrics, and score. The implementation details are "x" if the paper considers a balancing algorithm (**Ba**) and/or a preprocessing schedule (**Pp**).

Kaltwang et al. [3] propose a continuous pain estimation adopting Local Binary Patterns (LBP) and the Discrete Cosine Transform (DCT) as features. Then, 2-levels of Relevance Vector Machines (RVM) perform the prediction. The first one estimates the pain intensity for each element independently, while, the second calculates the pain intensity, considering the previous layer results. Zhao et al. [15] propose an Ordinal Support Vector Regression (OSVR) based on an SVR and an Ordinal Regression (OR). As a result, the OR establishes the temporal order while the SVR computes the intensity value. Each frame splits into five regions to extract two features per each: LBP and Gabor wavelet coefficients. Subsequently, a PCA is used to couple the feature vector of each area. For regression, they use linear kernel (L1) and quadratic kernel (L2) alongside the OSVR.

The majority of deep-learning approaches consider temporal rather than spatial analysis. Zhou et al. [16] propose a recurrent CNN (RCNN) to achieve a continuous pain estimation. To do so, they develop a preprocessing scheme, which includes: histogram equalization, face masking, and framing using facial landmarks and eye patches as references. Face images are flattened into 1D vectors and merge into a matrix, which is processed by the RCNN considering the last frame's label as output. Nasrollahi et al. [6] use a CNN-LSTM architecture to perform a 14-levels pain classification. They introduce a data balancing module by resolution variations over the less numerous classes. Then, a pre-trained VGG-16 architecture is fine-tuned to use it as a feature extractor. Finally, one LSTM layer performs the temporal analysis using the last frame's label as the sequence's label.

Rodriguez et al. [11] follow the same steps proposed by [[6]] but adding a preprocessing scheme like Zhou et al. [16]. Also, they prioritize the frontalization problem and develop a data augmentation module by landmark-based random deformations and vertical flipping. Frontalization attempts to solve the camera perspective error by estimating the projection matrix [1]. Data augmentation supports the less represented data and increases the training data. Finally, there are two considerations to make a fair comparison. First, the number of pain levels are proportional to the estimation accuracy. Second, the temporal analysis is much more complex, but it has greater medical acceptance than the stationary report.

3 Methodology

We use a standard pipeline for spatiotemporal analysis. First, we propose a preprocessing scheme that depends on the dataset conditions and the CNN requirements. Second, we do data balancing and data augmentation policies to offset unbalance. Regarding the architecture, we provide the implementation details alongside their explanations at each stage. Figure 1 illustrates our proposal overview.

3.1 Image Preprocessing

We consider three stages for pre-processing: (1) light normalization; (2) masking; and, (3) frontalization. Normalization bypasses the illumination problem and standardizes the input values. For masking, we use the convex hull algorithm over the facial landmarks given in the dataset. The frontalization matches the original landmarks with the frontal-view landmarks of a 3D-model[4] by computing the camera matrix and the projection matrix during the camera calibration. Thereupon, the projection matrix maps the original image to estimate the frontalized frame. Finally, the frontalized face undergoes a smooth symmetry process that fills the occluded parts using the opposite half of the face. Nonetheless, frontalization modifies and introduces new information or noises that could

[4] `https://github.com/dougsouza/face-frontalization`

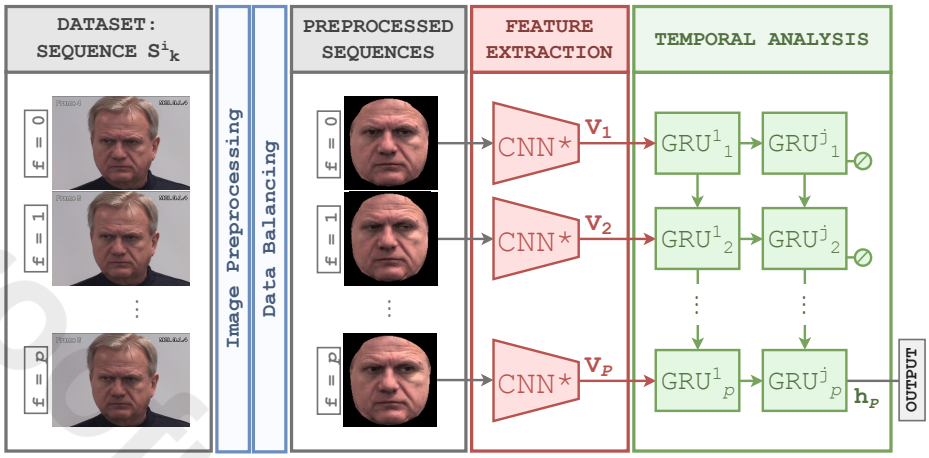

Fig. 1. Overview of our proposal. Given a sequence S_k, we extract the faces from each frame f that belongs to S_k. Then, a pre-trained CNN obtains the feature vector V_f from each face. Latter, an RNN establish the temporal correspondence between the feature vectors and the pain level. The RNN is composed of j layers of GRU layers units (GRU^j_p). The last frame's label (h_p) is the output of the whole sequence instead of its offline value.

interfere with the faces representation inside the model [1]. To surmount this problem, we define critical landmarks based-on the PSPI metric to compute the frontalization; these are the ones from the chin and cheeks contours; and, in the extremes of eyes, mouth, and eyebrows.

3.2 Data Balancing

The UNBC-McMaster Shoulder Pain database has a notorious unbalance at frames and patient distributions. For the temporal analysis, each patient has different amounts of videos, and each video has different sizes. To deal with the imbalance at frames and sequence level, we consider similar strategies to those raised by Nasrollahi et al. [6] and Rathee et al. [10]. Data augmentation includes rotations ($3°, 6°$ and $9°$), flipping and facial affine deformations. Finally, we introduce two sequential data generation policies: (1) sequences fragmentation; and, (2) overlapping.

Frames Balancing The frames balancing includes (1) downsampling the most represented classes; and, (2) a data augmentation policy for the least represented. The distribution matrix $M_{[p,l]}$ represents the data distribution for the patients $p \in \mathbb{P}$ and pain levels $l \in \mathbb{L}$. Then, we compute the scaling matrix $T_{[p,l]}$ which is formed by the scale factor τ of each element of $M_{[p,l]}$ concerning an established number of samples. The scale factor defines the policies to carry out and belongs to the $[0, \tau_{max}]$ range. So, if $\tau < 1$, downsampling is done with a probability of

τ, while if $\tau > 1$, data augmentation policies are applied with the same odds, except for the facial affine deformations. The facial affine deformations happen between 20% and 30% over the samples, including the generated ones.

Sequences Balancing The methods presented also work for sequence balancing by applying them to every frame of a sequence. However, these sequences are not the original ones because they must have the same size and have to represent a single pain stimulus. The initial sequences have painless dead-times both at the beginning and the end of the videos. First, the sequences split into 'pain' sections and 'painless' sections. Thereupon, a window runs along each section to create new subsequences. The subsequences obtained generates a distribution, over which we use the frame balancing reasoning.

3.3 Feature extraction

Feature extraction is the core of convolutional neural networks (CNN). Convolutional layers learn the main features of a dataset, then, fully connected layers correlate feature vectors with the outputs. For our case, we made a transfer learning procedure from `VGG_faces`[5] [7] to our VGG16 architecture. Figure 2 shows the `VGG_faces` model while Table 2 presents the model configuration. We apply a categorical cross-entropy loss to maximize the separation between classes; 50% of dropout probability; and, SGD optimization with a learning rate equal to 0.001 and momentum equal to 0.9. After evaluating the feature vectors, we opt to use the ttfc6 layer for the temporal analysis.

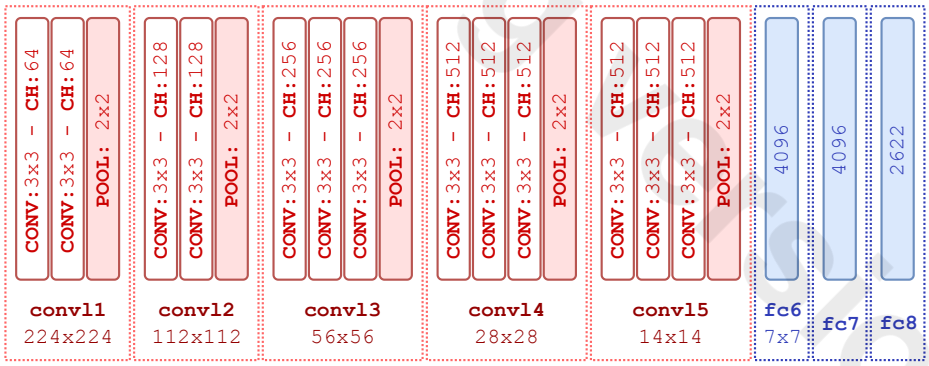

Fig. 2. Architecture `VGG_faces` raised by [7]. It is composed of 13 convolutional layers, five pooling layers and three fully connected layers. `CONV` and `POOL` are convolutional and max-pooling kernel sizes, respectively, while `CH` is the number of output channels. `fc` layers correlate the feature vectors to face classification tasks. `fc8` has 2622 output classes.

[5] `http://www.robots.ox.ac.uk/~vgg/software/vgg_face/`

Layers		Number of Filters	Input Size	Kernel Size	Str	Pad	Activation Function
Input	Image	1	224 × 224 × 3	-	-	-	ReLU
1	2 × `convl`	64	224 × 224 × 64	3 × 3	(1,1)	(1,1)	ReLU
	`MaxPool`	64	112 × 112 × 64	2 × 2	(2,2)	(0,0)	ReLU
3	2 × `convl`	128	112 × 112 × 128	3 × 3	(1,1)	(1,1)	ReLU
	`MaxPool`	128	56 × 56 × 128	2 × 2	(2,2)	(0,0)	ReLU
5	3 × `convl`	256	56 × 56 × 256	3 × 3	(1,1)	(1,1)	ReLU
	`MaxPool`	256	28 × 28 × 256	2 × 2	(2,2)	(0,0)	ReLU
8	3 × `convl`	512	28 × 28 × 512	3 × 3	(1,1)	(1,1)	ReLU
	`MaxPool`	512	14 × 14 × 512	2 × 2	(2,2)	(0,0)	ReLU
11	3 × `convl`	512	14 × 14 × 512	3 × 3	(1,1)	(1,1)	ReLU
	`MaxPool`	512	7 × 7 × 512	2 × 2	(2,2)	(0,0)	ReLU
14	`fc6`	4096	512	7 × 7	(1,1)	(0,0)	ReLU
15	`fc7`	4096	4096	1 × 1	(1,1)	(0,0)	ReLU
16	`fc8`	2622	4096	1 × 1	(1,1)	(0,0)	ReLU
Output	Prediction	2622	2622	1 × 1	(1,1)	(0,0)	*Softmax*

Table 2. `VGG_faces` configuration. From left to right: (1) layers description; (2) number of filters; (3) input size; (4) kernel size; (5) stride value *Str*; (6) padding value *Pad*; and, (7) activation function.

3.4 Temporal analysis

The temporal analysis is a complex problem which allows establishing correlations between the features from different states in a time series. For our case, pain estimation in a sequence depends on the pain expressed in each of its frames. Then, each sequence records to the pain stimulus effect fade in timespans. Based on this assumption, offline metrics were developed to evaluate each series as a whole. Nonetheless, the offline metrics are subjective, unlike the PSPI metric, which is physiognomic. To overcome this problem, we design the subsequence generation algorithm considering the last frame's label as the sequence's label. Thereby, the PSPI metric is used to label the new series.

GRUs are an alternative method to LSTMs to buffer the gradient vanishing problem. Similar to LSTMs, GRUs use gates to control its internal memory (h') and shared memory (h) along with each unit. These gates are the update gate (z_k) and the reset gate (r_k). Vizcarra et al. [13] argue that sequential-processing efficiency depends on its size. Thus, GRU simplicity reduces training time and favor short sequences processing. Figure 3 shows GRU schema in detail. Update gate z_k controls the influence of the previous state h_{k-1} and the current entry x_k over h_k and h'_k (Equation 1). Restart gate r_k defines the information must be reinforced from h_{k-1} and x_k in the unit memory h'_k (Equation 2). Equation 3 shows h'_k after r_k. $1 - z_k$ is used as an add-on to update h_{k-1} because z_k is close to 1. Finally, to compensate $1 - z_k$, z_k updates h'_k which adds new information to h_k (Equation 4).

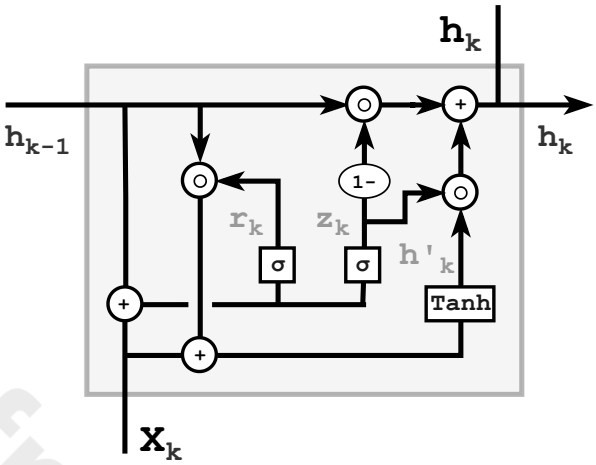

Fig. 3. Outline of a GRU. X_k and h_k are input and output of unit k, respectively. h_k is the shared memory that passes through all groups; h'_k is the internal memory; and, z_k and r_k are the update and reset gates, respectively. σ_g represents a sigmoid activation function while $tanh$ is a hyperbolic tangent; and, $\odot$ represents the Hadamard product.

$$z_k = \sigma_g(W_z X_k + U_z h_{k-1} + b_z) \tag{1}$$

$$r_k = \sigma_g(W_r X_k + U_r h_{k-1} + b_r) \tag{2}$$

$$h'_k = tanh(W_h X_k + U_h(h_{k-1} \odot r_k) + b_h) \tag{3}$$

$$h_k = h_{k-1} \odot (1 - z_k) + h'_k \odot z_k \tag{4}$$

4 Experiments and results

In this section, we present the results of each stage; also, the dataset description. To run experiments, we use a PC with the following settings: 3,6 GHz Intel Core i7 processor, 16 GB 3000 MHz DDR4 memory and NVIDIA GTX 1080ti. For training and testing, we use the Pytorch framework.

4.1 Dataset Description

The dataset has nearly 200 videos from 129 participants, 63 men, and 66 women, self-identified with chronic shoulder pain. Healthy arm moves are labeled as painless while the facial expressions obtained with the damaged arm are quantified using the Facial Action Coding System (FACS) of pain. The PSPI metric measures pain intensity at frame level by a linear combination of most relevant AUs.

Figure 4 shows some samples from the UNBC-McMaster database, while Table 3 presents the original data distribution per level. Besides, the dataset contains 66 facial landmarks per frame, which are calculated by an Active Appearance Model (AAM).

	PSPI Score															
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Original Samples	40029	2909	2351	1412	802	242	270	53	79	32	67	76	48	22	1	5
Total	**48398**															
Balanced Frames	500	500	500	500	500	500	500	500	500	500	500	500	500	380	16	80
Total	**6976**															
Balanced Sequences	300	300	300	300	300	300	300	300	300	300	300	300	300	300	16	80
Total	**6976**															

Table 3. Data distributions before and after balancing at both levels. The proposed policy for sequence balancing generates less data than frames balancing.

4.2 Analysis at frames level

For the frame-by-frame analysis, we fine-tune a pre-trained CNN due to pain classification is an end-to-end problem. The data split randomly into 80% as the training set and 20% as the testing set. Also, we use the one-subject-out strategy to measure the performance. Figure 5 presents the confusion matrix for the testing set, which is almost a diagonal matrix, although it has blocked at all levels. The best results are obtained in the best-represented levels, albeit data augmentation cushions this effect effectively. Table 4 summarizes all experiments carried out for pain recognition at frames level. The preprocessing and data augmentation stages have a highly significant contribution to the final results.

The results improve according to the processing complexity used; however, some levels show particular details, such as class 14, that has the least amount of data, so its representation is almost trivial. The camera perspective affects the sample representation due to the frontalization error increases between more occlusions exist. Table 5 shows the comparison of the result with the state-of-the-art in handcrafted features because most of those methods perform a frame-by-frame analysis.

4.3 Analysis at sequences level

Sequential analysis is a seq-to-end problem and has three key hyperparameters to tune: sequences resampling rate rr; sequences size ss; and, the number of recurrent layers nl. We can use the feature vectors from any of the fully-connected

Fig. 4. Examples of some sequences cuts from the UNBC-McMaster Pain Shoulder Archive. First row: the painless patient. Second row: each frame has a PSPI = 2. Third row: each frame has a PSPI = 3. Face deviations, as well as perspective deformation, can be seen.

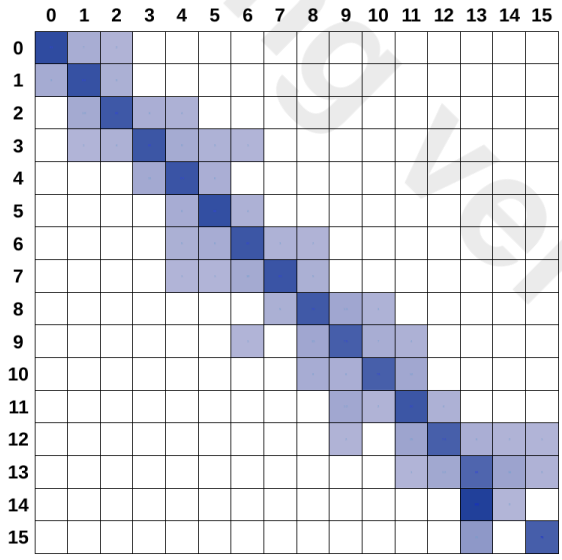

Fig. 5. Confusion matrix for 16-classes at frames level. The trend is mainly diagonal, which shows high precision.

	Frames level															
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Down	62.5	56.5	48.8	56.3	50.3	55.2	48.1	41.4	45.4	45.7	52.7	48.4	45.2	51.8	0	80
Average accuracy: 55.86%																
Prep	87.3	74.8	69.2	68.2	72.0	69.1	63.8	68.1	54.6	58.3	63.4	49.1	45.0	57.1	0	100
Average accuracy: 78.17%																
Prep Aug	91.2	85.6	78.6	83.1	82.5	85.1	83.2	81.1	78.4	73.9	72.4	84.1	74.3	69.9	**13.3**	76.2
Average accuracy: 80.44%																

Table 4. The comparison of result for different processing stages during the frame-by-frame analysis. We test our model using downsampled data (Down), preprocessed data (Prep) and preprocessed augmented data (Prep-Aug).

	Metrics				
	MAE	**MSE**	**PCC**	**ICC**	**ACC**
Zhao et al. [15]	0.81	-	0.60	**0.56**	-
Florea et al. [2]	-	1.18	0.55	-	-
Kaltwang et al. [3]	-	1.39	0.59	0.50	-
Rathee et al. [10]	-	-	-	-	**0.96**
Our	**0.475**	**0.592**	**0.723**	0.52	0.80

Table 5. Comparison of metrics at frames level

layers, although the `fc6` layer has a higher dimension and implies a better facial representation. Feature vectors feed each GRU unit as inputs, and the last frame's label defines the sequence's label. Table 6 shows some results of pain recognition at sequence level considering the following set of hyperparameters: $rr = 3$, $ss = 15$ and $nl = 2$. Data augmentation enables some levels to overcome the lack of data while decreasing the precision gap between classes. Table 7 shows the state-of-the-art comparison including our results.

5 Conclusions and future works

The pain analysis is very different at frames level that in sequence level, both at the application and processing. The frame-by-frame analysis of pain is done using only the spatial information, instead of considering the temporal correlation. The number of pain levels and its distribution affects the results significantly. Hence, the more degrees of pain, the higher the difficulty in classification. However, most confusions occur in very-close levels because of the closeness of their facial descriptors; as a result of this, the confusion matrix is block-diagonal.

The results show that the preprocessing stages achieved to separate the facial features of pain from the spatial and identity elements, but they can still be improved, especially the frontalization step. Data augmentation allows weight

	Sequential level															
	0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**
Down	71.9	63.9	66.9	64.9	66.6	74.1	64.7	36.8	81.8	-	80.0	78.6	74.1	-	-	-
Average accuracy: 69.73%																
Prep	85.1	78.2	75.6	69.9	74.9	88.2	74.1	61.7	86.8	-	84.4	78.2	77.3	-	-	-
Average accuracy: 78.27%																
Prep Aug	94.0	86.8	75.1	83.4	86.7	89.3	88.2	87.1	85.4	70.2	82.2	79.2	80.3	72.3	62.8	87.2
Average accuracy: 84.54%																

Table 6. The comparison of results for different processing stages during the sequential analysis. We test our model using downsampled data (Down), preprocessed data (Prep) and preprocessed augmented data (Prep-Aug).

	Metrics				
	MAE	**MSE**	**PCC**	**ICC**	**ACC**
Zhou et al. [16]	-	1.54	0.65	-	-
Nasrollahi et al. [6]	-	-	-	-	0.619
Rodriguez et al. [11]	0.5	0.74	**0.78**	0.45	-
Our	**0.487**	**0.615**	0.697	**0.60**	**0.84**

Table 7. Comparison of metrics at sequences level

each level result; hence, the average accuracy goes up. Finally, we overcome the-state-of-the-art metrics (accuracy, MAE, MSE, PCC, ICC) at both levels. In the future, we will analyze attention models to focus the learning process on the action units alongside visual interpretability tools.

References

1. Banerjee, S., Brogan, J., Krizaj, J., Bharati, A., Webster, B.R., Struc, V., Flynn, P.J., Scheirer, W.J.: To frontalize or not to frontalize: Do we really need elaborate pre-processing to improve face recognition? In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 20–29. IEEE (2018)
2. Florea, C., Florea, L., Vertan, C.: Learning pain from emotion: transferred hot data representation for pain intensity estimation. In: European Conference on Computer Vision. pp. 778–790. Springer (2014)
3. Kaltwang, S., Rudovic, O., Pantic, M.: Continuous pain intensity estimation from facial expressions. In: International Symposium on Visual Computing. pp. 368–377. Springer (2012)
4. Lucey, P., Cohn, J.F., Matthews, I., Lucey, S., Sridharan, S., Howlett, J., Prkachin, K.M.: Automatically detecting pain in video through facial action units. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **41**(3), 664–674 (2011)
5. Lucey, P., Cohn, J.F., Prkachin, K.M., Solomon, P.E., Matthews, I.: Painful data: The unbc-mcmaster shoulder pain expression archive database. In: Automatic Face

& Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on. pp. 57–64. IEEE (2011)
6. Nasrollahi, K., Telve, T., Escalera, S., Gonzalez, J., Moeslund, T.B., Rasti, P., Anbarjafari, G.: Spatio-temporal pain recognition in cnn-based super-resolved facial images. In: Video Analytics. Face and Facial Expression Recognition and Audience Measurement: Third International Workshop, VAAM 2016, and Second International Workshop, FFER 2016, Cancun, Mexico, December 4, 2016, Revised Selected Papers. vol. 10165, p. 151. Springer (2017)
7. Parkhi, O.M., Vedaldi, A., Zisserman, A., et al.: Deep face recognition. In: BMVC. vol. 1, p. 6 (2015)
8. Pedersen, H.: Learning appearance features for pain detection using the unbc-mcmaster shoulder pain expression archive database. In: International Conference on Computer Vision Systems. pp. 128–136. Springer (2015)
9. Raffaeli, W., Arnaudo, E.: Pain as a disease: an overview. Journal of pain research **10**, 2003 (2017)
10. Rathee, N., Ganotra, D.: A novel approach for pain intensity detection based on facial feature deformations. Journal of Visual Communication and Image Representation **33**, 247–254 (2015)
11. Rodriguez, P., Cucurull, G., Gonzalez, J., Gonfaus, J.M., Nasrollahi, K., Moeslund, T.B., Roca, F.X.: Deep pain: Exploiting long short-term memory networks for facial expression classification. IEEE transactions on cybernetics (99), 1–11 (2017)
12. Souza, J.B.d., Grossmann, E., Perissinotti, D.M.N., Oliveira Junior, J.O.d., Fonseca, P.R.B.d., Posso, I.d.P.: Prevalence of chronic pain, treatments, perception, and interference on life activities: Brazilian population-based survey. Pain Research and Management **2017** (2017)
13. Vizcarra, G., Mauricio, A., Mauricio, L.: A deep learning approach for sentiment analysis in spanish tweets. In: International Conference on Artificial Neural Networks. pp. 622–629. Springer (2018)
14. Williamson, A., Hoggart, B.: Pain: a review of three commonly used pain rating scales. Journal of clinical nursing **14**(7), 798–804 (2005)
15. Zhao, R., Gan, Q., Wang, S., Ji, Q.: Facial expression intensity estimation using ordinal information. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3466–3474 (2016)
16. Zhou, J., Hong, X., Su, F., Zhao, G.: Recurrent convolutional neural network regression for continuous pain intensity estimation in video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 84–92 (2016)